**Risk-Based Testing (RBT) Approaches**

**Risk-Based Testing (RBT)**

Risk-Based Testing (RBT) is a **test strategy that prioritizes testing efforts based on the risk level of different components** of the software. The goal is to **identify and test the most critical functionalities first**, ensuring efficient use of time and resources.

**Risk-Based Testing Approaches**

**1. Risk Identification**

* Identify **potential risks** that could impact the system.
* Consider factors like **business impact, probability of failure, and complexity**.

**2. Risk Assessment & Prioritization**

* **Assess the severity and likelihood of risks** using a risk matrix.
* Prioritize high-risk areas for extensive testing.

**3. Test Planning Based on Risk**

* Allocate **more test effort** to high-risk areas.
* Define **test types** (functional, security, performance) for each risk.

**4. Test Execution & Monitoring**

* Begin testing with the highest-risk components.
* Continuously monitor risk factors and update the test plan as needed.

**Example of Risk-Based Testing**

**Scenario: Banking Application Testing**

A bank is launching a new online banking system. The following functionalities are identified:

1. **User Login** (High Risk)
2. **Fund Transfers** (High Risk)
3. **Balance Inquiry** (Medium Risk)
4. **Transaction History** (Medium Risk)
5. **Profile Settings** (Low Risk)

**Risk Assessment:**

| **Functionality** | **Business Impact** | **Failure Likelihood** | **Risk Level** |
| --- | --- | --- | --- |
| User Login | High | High | Critical |
| Fund Transfers | High | High | Critical |
| Balance Inquiry | Medium | Medium | Moderate |
| Transaction History | Medium | Low | Low |
| Profile Settings | Low | Low | Low |

**Testing Strategy:**

* **Critical (High Risk)** – Perform extensive **functional, security, and performance testing** on **User Login** and **Fund Transfers**.
* **Moderate (Medium Risk)** – Perform **functional and regression testing** on **Balance Inquiry** and **Transaction History**.
* **Low Risk** – Perform **basic functional testing** on **Profile Settings**.

**Advantages of Risk-Based Testing**

Ensures **critical defects** are caught early.  
Optimizes **resource allocation** by focusing on high-risk areas.  
Improves **test efficiency** by reducing unnecessary testing.  
Helps in **decision-making** by balancing testing efforts with deadlines.

**Conclusion**

Risk-Based Testing (RBT) helps **focus testing efforts where they matter most**. By prioritizing **high-risk functionalities**, organizations can improve software quality while optimizing time and cost.